ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Lemont, IL 60439

Coordinated Platooning with Multiple Speeds[1]

Fengqiao Luo, Jeffrey Larson, and Todd Munson

Mathematics and Computer Science Division

Preprint ANL/MCS-P7064-0617

June 2017

[1]This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under contract number DE-AC02-06CH11357.

Coordinated Platooning with Multiple Speeds

Fengqiao Luo[1], Jeffrey Larson[2], and Todd Munson[2]

[1]Northwestern University, Dept. of Industrial Engineering and Management Science.
fengqiaoluo2014@u.northwestern.edu
[2]Argonne National Laboratory, Mathematics and Computational Science Division.
{jmlarson,tmunson}@anl.gov

June 26, 2017

Abstract

Vehicle platooning is a nascent fuel-saving technology. In a platoon, vehicles travel one after another with small intervehicle distances; trailing vehicles in a platoon save fuel because they experience less aerodynamic drag. This work presents a coordinated platooning model with multiple speed options that integrates scheduling, routing, speed selection, and platoon formation/dissolution in a mixed-integer linear program that minimizes the total fuel consumed by a set of vehicles while traveling between their respective origins and destinations. We are aware of no other integrated platoon routing approach that allows vehicles to select the speed at which they traverse network edges. The performance of this model is numerically tested on a grid network and the Chicago-area highway network. We find that the fuel-saving factor of a multivehicle system significantly depends on the time each vehicle is allowed to stay in the network; this time affects vehicles' available speed choices, possible routes, and the amount of time for coordinating platoon formation. For problem instances with a large number of vehicles, we propose and test a heuristic decomposed approach that applies a clustering algorithm to partition the set of vehicles and then routes each group separately. When the set of vehicles is large and the available computational time is small, the decomposed approach finds significantly better solutions than does the full model.

1 Introduction

Improving the fuel efficiency of vehicles is essential to increasing energy independence and decreasing greenhouse gas emissions. In 2012, approximately 64% of global crude oil was used by the transportation sector (The International Energy Agency, 2014). In 2015, nearly 75% of the U.S. petroleum consumption was by the transportation sector (U.S. Energy Information Administration, 2015). In 2014, nearly 26% of total greenhouse emissions came from the transportation sector (U.S. Environment Protection Agency, 2014); cars and light-duty trucks were responsible for 60% of these emissions, while medium-duty and heavy-duty vehicles produced 23%.

To help reduce fuel consumption, the U.S. government sets higher fuel efficiency standards for passenger cars and heavy-duty vehicles (Harrington and Krupnick, 2012). In response to these regulations, automakers incorporate various engine technologies, including direct fuel-injection, turbocharging, and deceleration fuel shut-off (Navigant Research, 2014); they have also developed hybrid, fuel-cell, and pure-electric vehicles (Chan, 2007; Pollet et al., 2012). Fuel-efficiency technologies have a large potential market because they help consumers save money when fuel prices are high and they help meet efficiency standards. Most important, improving fuel efficiency is a necessary step toward environmental sustainability.

Vehicle platooning is another such promising fuel-efficient technology that involves coordinating multiple vehicles to form a trainlike grouping of vehicles on the highway. Vehicles in the platoon drive the same speed with small intervehicle distances; see Figure 1, obtained from Daimler Innovation, for an example. (For safety and convenience, it is common to consider a maximum platoon length so that, for example, platoons will not block freeway exits.)

Trailing vehicles in a platoon can save fuel because they experience less aerodynamic drag than when driving individually. The fuel saving rate for a trailing vehicle depends on many factors. Two major factors are the speed and the intervehicle distance. Experiments show that trailing vehicles in a platoon traveling 80 km/h experience an average reduction in fuel use of 21% (resp. 16%) when the intervehicle distance is 10 m (resp. 16 m) (Bonnet and Fritz, 2000). When the speed is reduced to 60 km/h, the average reduction is 16% (resp. 10%). Additionally, when vehicles are in constant communication with each other, fuel consumption in cascade braking and accelerating due to the delay of human response can be greatly reduced. For these reasons, the massive application of vehicle platooning in metropolitan areas may help relieve traffic congestion (Fernandes and Nunes, 2012).

Active areas of platooning research are broad. Understanding the dependence of fuel reduction on the traveling speed and the intervehicle distance requires mechanical and aerodynamic analysis (Browned et al., 2004; Luo and Shladover, 2011; Nowakowski et al., 2011). Other research has focused on the technology required to enable vehicle-to-vehicle communication, thereby allowing vehicles to maintain a common speed and small intervehicle distances. The use of wireless communication and navigation systems, including dedicated short-range communication, adaptive cruise control, and GPS, have been studied (Luo and Shladover, 2011; Nowakowski et al., 2011). Research into developing control systems to help form stable platoons for safety and fuel-saving purposes has also been performed (Li et al., 2013; Ghasemi et al., 2015; Liang, 2016; Liang et al., 2013; Wang et al., 2012) along with studies of platoon-formation strategies under various road conditions and related communication protocols (Hobert, 2012).

In contrast to vehicles operating with on-board wireless technologies, other

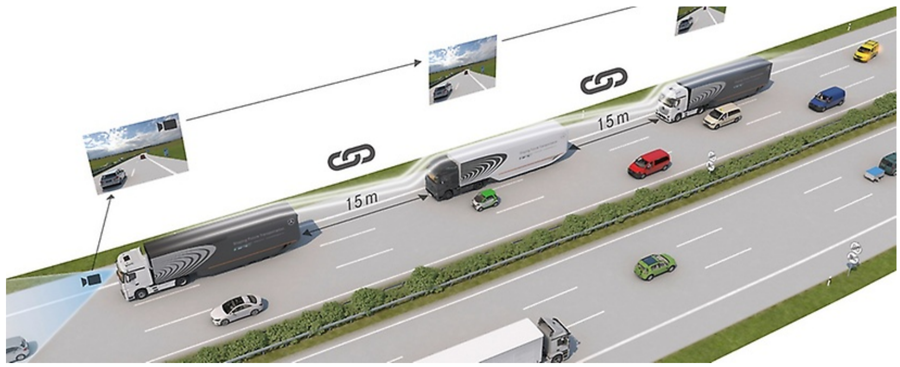


Figure 1: Illustration of vehicle platooning: Three heavy-duty vehicles form a platoon with 15 m intervehicle distances.

researchers have proposed paradigms where the transportation network itself plays a significant role in platooning. These intelligent transportation systems rely on monitors, sensors, short- and long-range communication protocols, and computational technologies to provide an integrated hardware and software base for realizing various platooning operations (Baskar et al., 2011).

While much research concerns designing and testing the tactical operations, designing headway control algorithms, and developing coordinating control systems for platoon formation and maintenance for the purposes of fuel-economy, stability, and safety (Alam et al., 2010; Bonnet and Fritz, 2000; Li et al., 2013; Shida and Nemoto, 2009; van de Hoef et al., 2015), our research focuses on the strategic operations, such as routing, speed selection, and scheduling of a set of vehicles with different origins and destinations in order to minimize the total fuel consumed by the group. These optimal routes, speeds, and schedules can then be sent to each vehicle, for example via GPS instructions.

Such coordinated routing has occasionally been studied. Baskar et al. (2013) propose a mixed-integer linear programming that minimizes the total travel time of a set of platoons. Larson et al. (2015) aim to minimize the total fuel consumption of concerned vehicles distributed in a transport network by routing and scheduling them to form or leave platoons. Larsson et al. (2015) have shown that finding an optimal routing and schedule for an arbitrary set of vehicles is NP-complete, but modeling techniques that exploit constraints common to many platooning networks have been shown to greatly decrease the time to solve such problems (Larson et al., 2016). Rather than centralized control, other researchers consider a distributed network of controllers that collect information from nearby vehicles and identify opportunities for these vehicles to share some subset of route edges in order to save fuel (Kammer, 2013; Liang, 2014).

The previous work on optimal platoon routing and coordination that we are aware of assumes that all vehicles traverse a given edge at the same speed. In this paper, we extend the model from Larson et al. (2016) to investigate a coordinated platooning model with multiple speed options (CPMS) for each vehicle. This model is presented in Section 2, and a range of experiments are described in Section 3. Since the fuel consumption of traveling a unit distance varies at different speeds, this modification provides vehicles more opportunities to save fuel and more flexibility in satisfying their arrival time requirements. In Section 4, we develop a heuristic decomposed method to find approximate solutions to centralized platooning problems with 1,000 vehicles by dividing the vehicles into groups. The grouping criterion is to minimize differences in the origins/destinations and starting/destination times among vehicles in the same subgroup; this is achieved by applying a clustering algorithm on a metric space of vehicles.

Table 1: Sets and parameters defining a CPMS model instance.

Set	Meaning	GAMS
V	vehicles to route	`V,W`
I	network nodes	`I,J,K`
$E \subseteq I \times I$	network edges	`E(V,I,J)`

Parameter	Meaning	GAMS
O_v	origin node for $v \in V$	`O(V)`
D_v	destination node for $v \in V$	`D(V)`
T_v^O	origin time for $v \in V$	`T_O(V)`
T_v^D	destination time for $v \in V$	`T_D(V)`
$B_{v,i}$	indicator of node i being the origin/destination of $v \in V$	`B(V,I)`
S_{ij}	set of speed options on edge $(i,j) \in E$	`S(I,J)`
$T_{i,j,s}$	time to take $(i,j) \in E$ with speed option $s \in S_{ij}$	`T(I,J,S)`
$C_{i,j,s}$	fuel cost for taking $(i,j) \in E$ with speed option $s \in S_{ij}$	`C(I,J,S)`
$H_{i,j}$	time to travel from i to j at maximum speed	NA
η_s	fuel saving rate per unit time under speed option s	`ETA(S)`

2 Coordinated Platooning with Multiple Speeds

We now present the CPMS model, which extends the coordinated platooning model from Larson et al. (2016) by providing vehicles speed options for each edge in the network. This additional freedom can decrease the total fuel consumption for two reasons. First, since a vehicle's fuel consumption per unit distance varies with respect to its speed, vehicles can drive at a more fuel-efficient speed. Second, allowing for speed options can increase the number platooning opportunities. (Introducing these decision variables increases the model complexity.) A GAMS implementation of this model and example problem data are available at

`http://www.mcs.anl.gov/~jlarson/Platooning`.

2.1 Model sets, parameters, and variables

In Table 1, we list the sets and parameters that are used to build the CPMS model. Because we implement our model in the GAMS modeling language (GAMS

Table 2: CPMS model variables.

Variable	Value Type	Meaning	GAMS
$f_{v,i,j,s}$	binary	1 if v travels on (i,j) with speed s	`f(V,I,J,S)`
$q_{v,w,i,j,s}$	binary	1 if v follows w on (i,j) with speed s	`q(V,W,I,J,S)`
$e_{v,i,j}$	non-negative	time v enters (i,j)	`time_e(V,I,J)`
$w_{v,i}$	non-negative	time v waits at i	`time_w(V,I)`

Table 3: Auxiliary parameters in the edge-based model.

GAMS Parameter	Meaning
`PE(V,I,J)`	1 if v can take (i,j)
`PQ(V,W,I,J)`	1 if v can follow w

Development Corporation, 2016), we also include the GAMS declarations as needed. These sets and parameters define a CPMS model instance.

In the CPMS model, the highway network is represented by a directed and connected graph $G(I,E)$, where I is the set of nodes in the graph and $E \subset I \times I$ is the set of directed edges. V is the set of all vehicles considered. The origin node O_v and the destination node D_v for a vehicle $v \in V$ are nodes in I. A vehicle's origin time T_v^O is defined as the earliest time that v can depart from O_v. (Vehicle v may wait at O_v until it departs.) Similarly, T_v^D is defined as the latest time by which v must reach D_v. The shortest-path, fastest-speed travel time between nodes $i,j \in I$ is denoted by $H_{i,j}$. For a problem to be feasible, $T_v^D \geq T_v^O + H_{O_v,D_v}$ must hold for all vehicles.

For any CPMS instance, we optimize by selecting the vehicles' routes, speeds, travel times, and by selecting whether vehicles are platooning on a given edge. We list these decision variables in Table 2. The variables f and q are binary variables, while e and w are positive reals.

We use modeling techniques to limit the declaration of some sets, decision variables, and constraints in order to reduce the model size in GAMS. For example, even with platooning opportunities, vehicles will never reach edges that are far away from their shortest path (Larson et al., 2016, Lemma 2.2). The model can safely restrict the edge set to include a dependence on the vehicle v and include only edges that each vehicle v can potentially travel on. That is, an edge is reachable for a vehicle v only if the vehicle reaches and traverses the edge and still reaches the destination within v's time constraints.

We list in Table 3 the auxiliary GAMS parameters used to restrict sets and constraints. In the auxiliary set `PE(V,I,J)`, a triplet (v,i,j) is set to 1 if the vehicle $v \in V$ can potentially travel on edge $(i,j) \in E$. Declaration of `E(V,I,J)` is restricted by the following GAMS statement.

```
E(V,I,J)$(PE(V,I,J)) = yes;
```

The number of edges in the model is greatly reduced by imposing this restriction. Topologically, the candidate edges of a vehicle are restricted to be a "narrow belt" of edges surrounding the shortest path of the vehicle.

Similarly, `PQ(V,W,I,J)` is set to 1 when vehicles $v, w \in V$ can feasibly platoon on the edge (i, j). Formally, the following inequality must hold:

$$\begin{aligned} &\max\left\{T_v^O + H_{O_v,i}, T_w^O + H_{O_w,i}\right\} + H_{i,j} \\ &\quad \leq \min\left\{T_v^D - H_{D_v,j}, T_w^D - H_{D_w,j}\right\}. \end{aligned} \tag{1}$$

This ensures that if v and w go from their respective origin nodes to node i, go through edge (i, j) simultaneously at the fastest allowable speed, and go from node j to their respective destination nodes, then they must arrive by their deadlines.

The use of `PQ` and `PE` to reduce the model variables and constraints is essential to solving real-world platooning instances with more than 10 vehicles.

2.2 Assumptions

We make the following assumptions in our model. First, each vehicle is allowed to drive at any speed S_{ij} on any arc $(i, j) \in E$, where S_{ij} (Table 1) is the feasible set of speeds that a vehicle can drive at on edge (i, j). Second, we ignore any traffic conditions that may affect the vehicle speed. Third, for the considered vehicles, the coordinated platooning problem we investigate is formulated and solved before the earliest entering time of the vehicles in the network, that is, $\min\limits_{v \in V} T_v^O$. Fourth, vehicles are numbered so that vehicles with smaller smaller indices lead platoons. Note that this assumption can be embedded in the definition of `PQ(V,W,I,J)` such that a quadruplet (v, w, i, j) can be nonzero only if $v < w$. This assumption reduces some unnecessary symmetry and does not affect the total fuel consumption. (Vehicles can easily be reordered in postprocessing if necessary.)

2.3 Objective function

The collective amount of fuel used is

$$\sum_{v,i,j} \sum_{s \in S_{ij}} C_{i,j,s} \left(f_{v,i,j,s} - \eta_s \sum_{w} q_{v,w,i,j,s} \right). \tag{2}$$

$C_{i,j,s}$ is the amount of fuel used by a vehicle to traverse edge (i, j) at speed s, and η_s is the fraction of fuel saved by platooning at speed s. One can consider $C_{i,j,s}$ to be a vehicle-dependent value without increasing the number of decision variables, but we do not do so here. The first term in (2) is the fuel consumption of vehicles driving without another vehicle in front of them; the second term is the amount of saved fuel due to platooning. If desired, one can also include a penalty in the objective for the time vehicles spend waiting. To study the upper bound on possible platoon savings, we do not include wait times.

2.4 Multispeed coordinated platooning model constraints

We now declare the constraints for the CPMS model.

- Each vehicle can have at most one speed per edge.

$$\sum_{s\in S_{ij}} f_{v,i,j,s} \leq 1 \qquad \forall v\in V,\ (i,j)\in E. \tag{3}$$

- Node outflows must equal inflows:

$$\begin{aligned}&\sum_{j:(i,j)\in E}\sum_{s\in S_{ij}} f_{v,i,j,s} = \sum_{j:(j,i)\in E}\sum_{s\in S_{ji}} f_{v,j,i,s} + B_{v,i}\\ &\forall v\in V,\ i\in I,\end{aligned} \tag{4}$$

where $B_{v,i}$ is 1 if $i = O_v$, -1 if $i = D_v$, and 0 otherwise.

- When platooning, enter times must be equal.

$$\begin{aligned}-\mathbb{M}_{1l}\left(1-\sum_{s\in S_{ij}} q_{v,w,i,j,s}\right) &\leq e_{v,i,j} - e_{w,i,j}\\ &\leq \mathbb{M}_{1r}\left(1-\sum_{s\in S_{ij}} q_{v,w,i,j,s}\right)\\ \forall v,w\in V,\ (i,j)\in E,\ v>w.\end{aligned} \tag{5}$$

- A vehicle can follow at most one other vehicle.

$$\sum_{w:w<v} q_{v,w,i,j,s} \leq 1 \quad \forall v\in V,\ (i,j)\in E,\ s\in S_{ij}. \tag{6}$$

- A vehicle can follow only one vehicle in its platoon.

$$\begin{aligned}&\sum_{w:w>v} q_{w,v,i,j,s} \leq 1\\ &\qquad \forall v\in V,\ (i,j)\in E,\ s\in S_{ij}.\end{aligned} \tag{7}$$

- Platooning requires flow for both leader and follower.

$$\begin{aligned}2q_{v,w,i,j,s} &\leq f_{v,i,j,s} + f_{w,i,j,s}\\ \forall v,w\in V,\ v>w,&\ (i,j)\in E,\ s\in S_{ij}.\end{aligned} \tag{8}$$

- T_v^O plus w_{v,O_v} is the origin enter time.

$$\begin{aligned}-\mathbb{M}_{2l}\Big(1-\sum_{s\in S_{O_v,j}} f_{v,O_v,j,s}\Big) &\leq e_{v,O_v,j} - T_v^O - w_{v,O_v}\\ &\leq \mathbb{M}_{2r}\Big(1-\sum_{s\in S_{O_v,j}} f_{v,O_v,j,s}\Big)\\ \forall v\in V,\ j\in \big\{j'\in I : (O_v,j')\in E\big\}.\end{aligned} \tag{9}$$

- T_v^D is the final enter time plus the time required to travel the final edge plus w_{v,D_v}.

$$
\begin{aligned}
&- \mathbb{M}_{3l}\Big(1 - \sum_{s\in S_{i,D_v}} f_{v,i,D_v,s}\Big) \\
&\le T_v^D - e_{v,i,D_v} - w_{v,D_v} - \sum_{s\in S_{i,D_v}} T_{i,D_v,s} f_{v,i,D_v,s} \\
&\le \mathbb{M}_{3r}\Big(1 - \sum_{s\in S_{i,D_v}} f_{v,i,D_v,s}\Big) \\
&\forall v \in V,\ i \in \big\{i' \in I :\ (i', D_v) \in E\big\}.
\end{aligned}
\tag{10}
$$

- Enter times at intermediate nodes must match the time increment due to travel and waiting time.

$$
\begin{aligned}
&- \mathbb{M}_{4l}\Big(2 - \sum_{s\in S_{ij}} f_{v,i,j,s} - \sum_{s'\in S_{jk}} f_{v,j,k,s'}\Big) \\
&\le e_{v,j,k} - e_{v,i,j} - w_{v,j} - \sum_{s\in S_{ij}} T_{i,j,s} f_{v,i,j,s} \\
&\le \mathbb{M}_{4r}\Big(2 - \sum_{s\in S_{ij}} f_{v,i,j,s} - \sum_{s'\in S_{jk}} f_{v,j,k,s'}\Big) \\
&\forall v \in V,\ (i,j),(j,k) \in E,\ j \neq O_v, D_v.
\end{aligned}
\tag{11}
$$

- If there is no flow, the enter time must be zero.

$$
e_{v,i,j} \le \mathbb{M}_5 \sum_{s\in S_{ij}} f_{v,i,j,s} \quad \forall v \in V,\ (i,j) \in E. \tag{12}
$$

- If there is no flow, the wait time must be zero.

$$
\begin{aligned}
&w_{v,i} \le \mathbb{M}_6 \left(\sum_{i,j} \sum_{s\in S_{ij}} \sum_{s'\in S_{ji}} f_{v,i,j,s} + f_{v,j,i,s'}\right) \\
&\forall v \in V,\ i \in I.
\end{aligned}
\tag{13}
$$

We now briefly explain these constraints. Constraint (4) is the route planning constraint for each vehicle; (5) ensures time consistency of two vehicles that are platooning on an edge; and (6) and (7) define the leader-follower matching rule. In other words, constraints (6) and (7) ensure that every trailing vehicle in a platoon is set to follow the lead vehicle but not other vehicles in its platoon. Constraint (8) ensures speed consistency for two vehicles in the same platoon on an edge. Note that vehicles are allowed to wait at intermediate nodes in the CPMS model. In the numerical experiments, however, we limit the places where waiting can occur, experiments (Section 3). The remaining constraints ensure that each vehicle's traveling and waiting times are consistent.

Note that a big-M parameter is involved in our formulation. One possible choice for $\mathbb{M}$ in constraints (5), and (9)-(13) is

$$\mathbb{M} = \max_{v} \left\{T_v^D\right\}.$$

However, such an $\mathbb{M}$ does not need to be set uniformly. Instead, we can set different values of $\mathbb{M}$ for these constraints to tighten the formulation and tune $\mathbb{M}$ according to the specific instance that needs to be solved. Proposition 2.1 gives a way of setting $\mathbb{M}$, and we implement these $\mathbb{M}$-values in the GAMS model for numerical experiments. (Mixed-integer linear programming solvers often tighten $\mathbb{M}$ further during preprocessing.)

Proposition 2.1. *In the* CPMS *model, the following ways of setting* $\mathbb{M}$ *for each related constraint independently is valid for any feasible solution.*

(a) *In* (5)*, for the constraint induced by* $v, w \in \mathtt{V}$ *and* $(i,j) \in E$ *such that* $(v,w,i,j) \in \mathtt{PQ(V,W,I,J)}$*, we can set* $\mathbb{M}_{1r} = (T_v^D - \min_{s \in S_{ij}} T_{ijs} - H_{j,D_v})^+$*. and* $\mathbb{M}_{1l} = (T_w^D - \min_{s \in S_{ij}} T_{ijs} - H_{j,D_w})^+$ *for each* $v, w \in V$*,* $v > w$*,* $(i,j) \in E$*;*

(b) *In* (9)*, we can set* $\mathbb{M}_{2r} = 0$ *and* $\mathbb{M}_{2l} = T_v^D - H_{O_v,D_v}$ *for each* $v \in V$*.*

(c) *In* (10)*, we can set* $\mathbb{M}_3 l = 0$ *and* $\mathbb{M}_{3r} = T_v^D$ *for each* $v \in V$*.*

(d) *In* (11)*, we can set* $\mathbb{M}_{4r} = (T_v^D - \min_{s \in S_{j,k}} T_{j,k,s} - H_{k,D_v})^+$ *for each* $v \in V$ *and* $(i,j) \in E$ *and* $\mathbb{M}_{4l} = (T_v^D - H_{j,D_v})^+$ *for each* $v \in V$ *and* $j \in \left\{j' \in I : (i,j'),(j',k) \in E\right\}$*.*

(e) *In* (12)*, we can set* $\mathbb{M}_5 = (T_v^D - \min_{s \in S_{ij}} T_{i,j,s} - H_{j,D_v})^+$ *for each* $v \in V$*.*

(f) *In* (13)*, we can set* $\mathbb{M}_6 = (T_v^D - T_v^O - H_{O_v,i} - H_{i,D_v})^+$ *for each* $v \in V$*,* $i \in I$*.*

In all cases, $(x)^+ = \max\{0, x\}$*.*

Proof. We prove only part (d); the proofs for the other parts are similar. For $\mathbb{M}_{4r}$, we require

$$e_{v,j,k} - e_{v,i,j} - w_{v,j} - \sum_{s \in S_{ij}} T_{i,j,s} f_{v,i,j,s} \leq \mathbb{M}_{4r}. \tag{14}$$

We have three cases for a feasible solution.

1. Vehicle v does not pass (j,k): Then $e_{v,j,k} = 0$, and hence $\mathbb{M}_{4r} = 0$ satisfies the requirement.

2. Vehicle v passes (j,k) and (i,j): Then we have $e_{v,j,k} - e_{v,i,j} - w_{v,j} - \sum_{s \in S_{ij}} T_{i,j,s} f_{v,i,j,s} = 0$, and hence $\mathbb{M}_{4r} = 0$ satisfies the requirement.

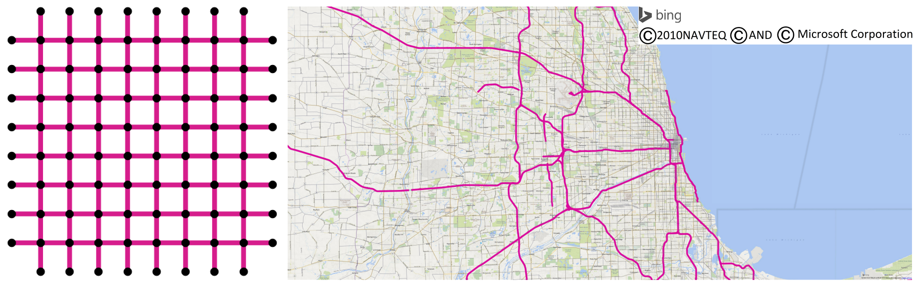


Figure 2: Networks considered: a grid (left) and the Chicago-area highways (right).

3. Vehicle v passes (j,k) but does not pass (i,j): In this case, we have $e_{v,i,j}=0$ and $f_{v,i,j,s}=0$. Then (14) becomes $e_{v,j,k}-w_{v,j}\le \mathbb{M}_{4r}$. Since $e_{v,j,k}+\min_{s\in S_{j,k}} T_{j,k,s}+H_{k,D_v}\le T_v^D$ and $w_{v,j}\ge 0$, $\mathbb{M}4r=(T_v^D-\min_{s\in S_{j,k}} T_{j,k,s}-H_{k,D_v})^+$ satisfies the requirement.

For $\mathbb{M}_{4l}$, we require

$$e_{v,i,j}+w_{v,j}+\sum_{s\in S_{ij}} T_{i,j,s} f_{v,i,j,s}-e_{v,j,k}\le \mathbb{M}_{4l}. \tag{15}$$

Similarly, we have three cases for a feasible solution

1. The vehicle v does not pass either (i,j) or (j,k). Then (15) becomes $w_{v,j}\le \mathbb{M}_{4l}$. Since $T_v^O+H_{O_v,j}+w_{v,j}+H_{j,D_v}\le T_v^D$, $\mathbb{M}_{4l}=(T_v^D-T_v^O-H_{O_v,j}-H_{j,D_v})^+$ satisfies the requirement.

2. The vehicle v passes (i,j), but does not pass (j,k). Then $e_{v,j,k}=0$. Since $e_{v,i,j}+\sum_{s\in S_{ij}} T_{i,j,s}f_{v,i,j,s}+w_{v,j}+H_{j,D_v}\le T_v^D$, $\mathbb{M}_{4l}=(T_v^D-H_{j,D_v})^+$ satisfies the requirement.

3. The vehicle v passes (j,k), but does not pass (i,j). Then (15) becomes $w_{v,j}-e_{v,j,k}\le \mathbb{M}_{4l}$. Since $w_{v,j}\le T_v^D-T_v^O-H_{O_v,j}-H_{j,D_v}$ and $e_{v,j,k}\ge T_v^O+H_{O_v,j}$, $\mathbb{M}_{4l}=(T_v^D-2T_v^O-2H_{O_v,j}-H_{j,D_v})^+$ satisfies the requirement.

4. The vehicle v passes both (i,j) and (j,k). Then $\mathbb{M}_{4l}=0$ satisfies the requirement.

This completes the proof. $\square$

3 Numerical Experiments

We test the computational performance of the CPMS model on a grid network and a representation of the Chicago-area highway network. From these results, we gain insight into the collective fuel savings that can be achieved with the coordinated platooning technology compared with each vehicle traversing its shortest path separately.

The considered networks are shown in Figure 2. The Chicago network is simplified by preprocessing the graph. In particular, nodes and edges are removed that cannot be used in any feasible solution, such as nodes that have no incoming edges and are not the origin node for some trip. Reductions also include replacing a node j with one incoming edge (i, j) and one outgoing edge (j, k) with a single edge (i, k) when node j is neither the origin nor the destination for any vehicle. If edge (i, k) already exists, then we keep the edge with the smallest fuel cost. If we assume there is no cost for waiting at a node, these modifications have no impact on the potential to platoon. When we have multiple speeds, however, the preprocessing does affect the fuel consumption savings that could be achieved. That is, we may reduce fuel consumption further by selecting different speeds along the edges in the original path as long as we reach the platooning formation point in time to join the platoon. These additional savings can easily be recovered during a postprocessing phase when the edges are disaggregated. A final part of the preprocessing evaluates the shortest paths from the origin to the destination and eliminates paths that cannot be used to reach the destination given the arrival deadline. We note that there may be cases where the same platoons are not selected in the CPMS model using the reduced and the original network. Using the original Chicago highway network, however, quickly results in an intractable instance.

3.1 Numerical setup

We first consider a 50-vehicle system with their origin/destination nodes randomly distributed throughout the network. Origin/destination nodes in the grid network are randomly generated. For the Chicago network, the origin/destination pairs are drawn uniformly from the 100 most common routes in the POLARIS (Auld et al., 2016) simulation of the Chicago highway network. We investigate two sets of speed parameters. The first allows two speed options while the second allows five speed options. All speed options are available for all vehicles on all edges. Note that speed is not directly correlated with fuel efficiency. Studies show that the most fuel-efficient speed is about $55 \sim 60$ mph, and fuel efficiency decreases as the speed increases and decreases. Detailed information about our two settings is given in Table 4 and Table 5, respectively. Note that we order the speeds so that s_i is greater than s_{i+1} (s_1 corresponds to the fastest speed). The numerical dependence of miles per gallon on the speed is based on U.S. Department of Energy (n.d., 2011). Using the results from Bonnet and Fritz (2000), a reasonable fuel-saving factor of follower vehicles in a platoon is about 0.15 at high speed and 0.1 at low speed. Therefore, we set set $\eta = 0.15$ for the speed at 75 miles/h and $\eta = 0.1$ for the speed at 50 miles/h.

Origin times T_v^O for each vehicle are drawn uniformly from $[0, 100]$, and destination times T_v^D are set to

$$T_v^D = T_v^O + (1 + P)H_{O_v,D_v}, \tag{16}$$

where H_{O_v,D_v} is the minimal time required to go from O_v to D_v using the shortest path at the maximum speed and P is the pause time ratio that describes how much time vehicle v can stay in the network to wait for other vehicles relative to its shortest-path traveling time. From the CPMS model, each vehicle can utilize its pause time to wait for other vehicles to form platoons at certain nodes in the network, to travel at slower speeds, or to perform some combination of the two in order to save fuel.

In reality, vehicle routes are given only T_v^O and T_v^D directly, and the pause time must be inferred. Hence P may be different for different vehicles in real-world problems, but we simplify its setting for our experiments. For a given network and speed setting, we construct 21 instances by setting P uniformly for every vehicle to be $[0, 0.1, \ldots, 2.0]$, respectively. In summary, our instances are specified by the following factors.

$$\begin{bmatrix}\text{Chicago}\\ \text{vs.}\\ \text{grid}\end{bmatrix} \otimes \begin{bmatrix}\text{2 speeds}\\ \text{vs.}\\ \text{5 speeds}\end{bmatrix} \otimes \left[P \in \{0, 0.1, \ldots, 2.0\}\right].$$

3.2 Analysis of computational times and solution quality

To analyze the performance of the CPMS model, we implemented it in GAMS and solved the aforementioned problem instances using Gurobi (Gurobi Optimization, Inc., 2017) with one thread. Gurobi was set to stop when the relative optimality gap was 0.05% or when the computational time reached one hour. The objective value after five minutes of computational time was also recorded. The CPMS model and example data are freely available at

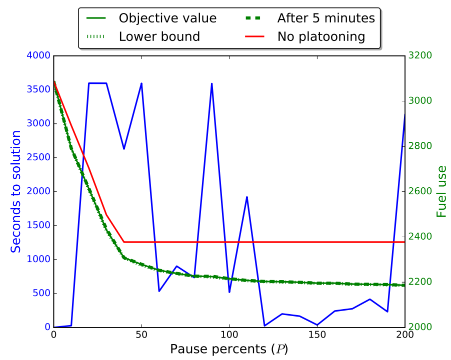
`http://www.mcs.anl.gov/~jlarson/Platooning`.

The objective values (total fuel consumption) and computational times for a range of pause percentages P are plotted in Figures 3(a)–3(b) (resp. Figures 3(c)–3(d)) corresponding to the Chicago highway (resp. grid) network. Also shown is the baseline case where vehicles never platoon but only adjust their speeds in order to minimize their fuel use.

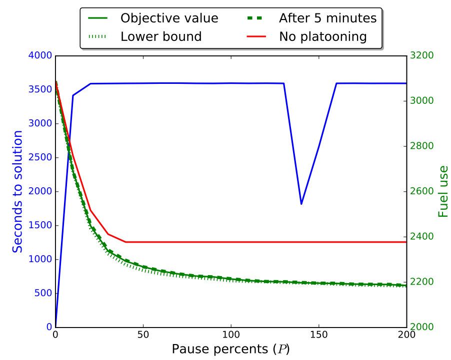
Of the Chicago-network instances, 45% do not solve within a one-hour time limit, while 92% of grid-network instances reach this limit. For the 2-speed setting, most Chicago-network instances are finished within thirty minutes, while most grid-network instances reach the one-hour limit. This may seem counterintuitive given that the Chicago network is much larger than the grid network.

Table 4: Parameters of two-speed setting used in numerical experiments.

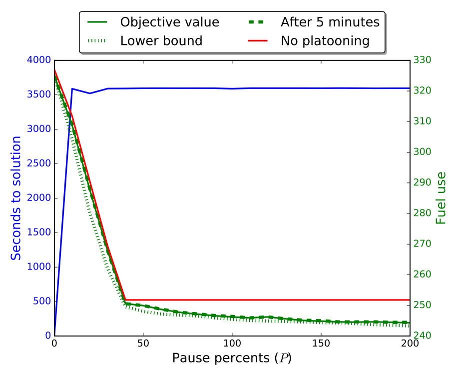
Speed Options	s_1	s_2
miles per hour	75	55
time cost per unit distance	1.00	1.36
fuel cost per unit distance	1.00	0.77
platooned fuel saving rate η	0.15	0.10



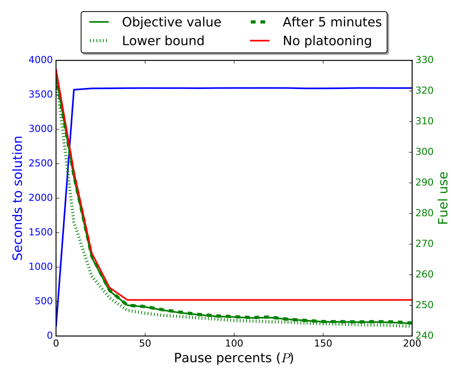
((a)) Chicago network, 2 speeds



((b)) Chicago network, 5 speeds



((c)) Grid network, 2 speeds



((d)) Grid network, 5 speeds

Figure 3: Computational time (blue line) and solution quality (green and red lines) for the CPMS model of the Chicago highway network as the willingness to wait increases. The solid green line is the objective value at termination (with the runtime limited to one hour). The dotted green line is the objective lower bound at the termination. The dashed green line is the objective value after five minutes. The solid red line represents the total fuel consumption when platooning is not allowed.

We believe the grid network to be the more difficult test case because of the existence of many shortest paths between most pairs of vertices. Coordinating platoons therefore requires significant exploration of the search space. Unsurprisingly, the 5-speed cases take significantly more computational time than do the 2-speed instances.

If the pause ratio P is zero, the problem instances are quickly solved because many decision variables are forced: by (16), vehicles must travel on a shortest path at the fastest speed. The computational time quickly increases as P increases from zero. This is especially pronounced in Figure 3(b) and Figures 3(c)–3(d).

Even though Figure 3 shows that one hour is often not enough to reduce the relative optimality gap below 0.05%, the solution after five minutes has nearly the same objective value as the solution after one hour. For the Chicago-network (resp. grid-network), instances, the average relative optimality gap over all instances is 0.16% (resp. 0.61%) and the maximum optimality gap is 0.73% (resp. 5.3%). These indicate that Gurobi finds good solutions to nearly all CPMS instances considered in a short amount of time. Therefore, we believe the CPMS model is suitable for practical use even when a central coordinator wishes to quickly adjust routes and schedule in response to changing traffic patterns.

3.3 Analysis of fuel-saving performance

The pause time P can be interpreted as the amount of time vehicles are willing to deviate from their shortest path times. When $P = 0$, the solution of the CPMS model routes are the union of shortest paths for each vehicle, and vehicles do not form platoons unless they meet another vehicle coincidentally. Hence, the $P = 0$ fuel consumption is the largest among all cases of pause time, and the computational time is small. As the pause time increases, opportunities for platooning appear and opportunities to use fuel-efficient speeds appear, as well; hence the corresponding fuel consumption decreases.

When $P > 0$, the amount of fuel saved (versus the fuel consumption at $P = 0$) comes from vehicles using slower speeds and forming platoons. To understand their relative contributions for each value of P, we also compute the fuel consumption of each vehicle traveling on its shortest path without platooning; these values are represented by the red curve in Figures 3(a)–3(b) and Figures 3(c)–3(d). For both platooning-forbidden instances and platooning-allowed instances, vehicles do not have to travel at the fastest speed when P

Table 5: Parameters of five-speed setting used in numerical experiments.

Speed Options	s_1	s_2	s_3	s_4	s_5
miles per hour	75	70	65	60	55
time cost/distance	1.00	1.07	1.15	1.25	1.36
fuel cost/distance	1.00	0.93	0.84	0.79	0.77
platn-fuel saving rate η	0.15	0.14	0.13	0.11	0.10

is sufficiently large. This strategy is reflected from Figures 3(a)–3(b) and Figures 3(c)–3(d) by the decreasing trend of the red curve and the solid green curve. For platooning-forbidden instances, after P is greater than the threshold that allows every vehicle traveling at the slowest speed, no further fuel reduction is possible. This situation occurs when the red curve becomes flat in the above figures; this threshold occurs when P is approximately 0.4. We see that significant savings can be incurred when vehicles travel at slower speeds.

The decrease in fuel use for P between 0 and 0.4 is faster for the 5-speed instances for both networks. This arises from vehicles having more speed options, and hence vehicles can drive at intermediate speeds in order to save fuel but still arrive at their destinations on time, as P deviates from 0.

We define two quantities γ_{speed} and γ_{platoon} that reflect the ratio of fuel saving due to the speed-choice strategy and the platooning strategy, respectively. They are functions of the pause ratio, and their expressions are given as follows:

$$\gamma_{\text{speed}}(P) = \frac{F^{\text{forb}}(0) - F^{\text{forb}}(P)}{F^{\text{forb}}(0)},$$
$$\gamma_{\text{platoon}}(P) = \frac{F^{\text{forb}}(P) - F^{\text{allow}}(P)}{F^{\text{forb}}(0)},$$

where quantities $F^{\text{forb}}(P)$ and $F^{\text{allow}}(P)$ respectively represent the fuel consumptions of platooning-forbidden (red curves) and platooning-allowed (solid green curves) modes at pause ratio P. Both γ_{speed} and γ_{platoon} increase as the pause ratio increases, and they reach their upper limits once the pause ratio is large enough. Note that we must always have $\gamma_{\text{platoon}} < \eta$, since η is the physical limit of the fuel-saving factor of a platooned vehicle. For numerical instances with five speeds and $P = 2.0$ (the best fuel saving instances in our study), we have $\gamma_{\text{speed}} \approx 23.0\%$ and $\gamma_{\text{platoon}} \approx 6.2\%$ for the Chicago-area highway network and $\gamma_{\text{speed}} \approx 22.5\%$ and $\gamma_{\text{platoon}} \approx 2.3\%$ for the grid network. The data shows that γ_{platoon} for the Chicago-area highway network is considerably greater than that for the grid network. The reason is possibly that the origin/destination pairs for the Chicago-area highway network are taken from the most commonly traveled routes. Therefore, significant overlap in vehicle routes is possible.

4 Heuristic Decomposition for Large-Scale Problems

Even for the 50-vehicle system with five speed options, the CPMS model involves 56,970 binary decision variables for the Chicago-area highway network; after one hour of computational time, the relative optimality gap is approximate 0.7% for some problem instances. For a 100-vehicle system, the number of binary variables increases to 780,610; this growth makes us believe that generating the instances of the CPMS model for hundreds of vehicles will be prohibitively expensive.

Of course, two vehicles are unlikely to form a platoon somewhere in an optimal strategy when their respective origin nodes, destination nodes, origin times, and destination times differ greatly. While this situation may be captured by the definition of `PQ` when (1) holds, we are further inspired to decompose large problem instances. The essential idea is to define a metric space on the set of vehicles and apply a clustering algorithm on this metric space in order to partition the set of vehicles. We then can solve a smaller CPMS model for each group independently. We define the metric

$$d(v_1, v_2) := \Big(dist^2(O_{v_1}, O_{v_2}) + dist^2(D_{v_1}, D_{v_2}) \\ + (T^O_{v_1} - T^O_{v_2})^2 + (T^D_{v_1} - T^D_{v_2})^2 \Big)^{1/2} \tag{17}$$

for all $v_1, v_2 \in V$, where $dist(i, j)$ is the distance between nodes i and j in the undirected graph extension of the considered road network. The metric (17) measures the similarity between two vehicles' routes in space and time. To better achieve this task, we unify the scale of the distance and time for evaluating $d(\cdot, \cdot)$.

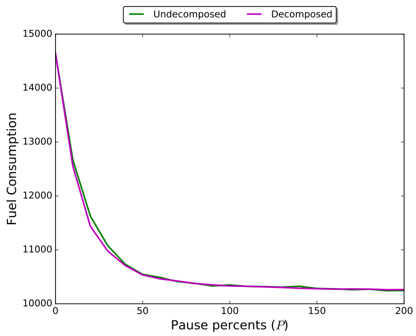
The clustering method used to decompose the set of vehicles is given in Algorithm 1, which is based on a modification of a K-sets algorithm of Chang et al. (2016) that includes a restriction on the size of each cluster. This clustering algorithm involves using the triangular distance.

Definition 4.1. *In a metric space, the* triangular distance *from a point x to a set S, denoted by $\Delta(x, S)$, is defined as*

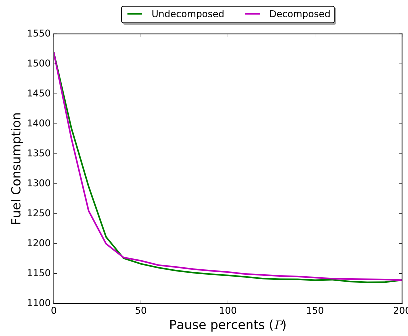
$$\Delta(x, S) = \frac{1}{|S|^2} \sum_{z_1 \in S} \sum_{z_2 \in S} \Big(d(x, z_1) + d(x, z_2) - d(z_1, z_2) \Big). \tag{18}$$

Algorithm 1 K-set clustering algorithm on the metric space of vehicles

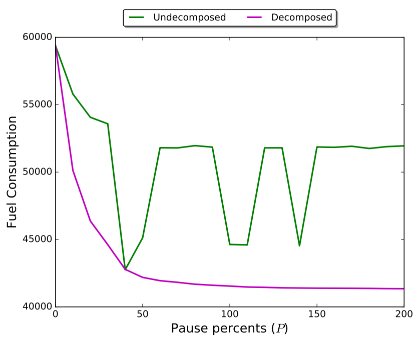
1: **Input**: A vehicle set $V = \{v_1, v_2, \ldots, v_n\}$, metric d, a number of clusters K, and a maximum size L of each cluster satisfying $KL > |V|$.
2: **Output**: A partition $\{S_1, \ldots, S_K\}$ of V with $|S_i| \le L$.
3: **Initialization**: Choose arbitrarily K disjoint nonempty sets $S_1, \ldots, S_K$ that partition V.
4: **for** $i = 1, 2, \ldots, n$ **do**
5:     Compute $\Delta(v_i, S_k)$ for each set S_k using (18).
6:     Find $k^* = \underset{\{k \in [K] : |S_k| < L\}}{\operatorname{argmin}} \Delta(x_i, S_k)$.
7:     **if** $x_i \notin S_{k^*}$ **then**
8:         $S_{k^*} \leftarrow S_{k^*} \cup \{x_i\}$.
9:     **end if**
10: **end for**
11: **Go to** Line 4 until there is no further change.



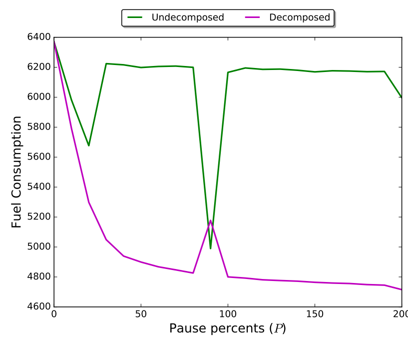
((a)) Chicago network with 250 vehicles



((b)) Grid network with 250 vehicles



((c)) Chicago network with 1,000 vehicles



((d)) Grid network with 1,000 vehicles

Figure 4: Numerical comparison between the decomposed and undecomposed approaches applied to a 250-vehicle system with the five speed settings (a) and (b), and to a 1,000-vehicle system with five speed settings (c) and (d). (The undecomposed approach is unable to identify a lower bound in one hour for the 1,000-vehicle instances.)

We perform further numerical experiments on CPMS to test the performance of the decomposed approach versus the undecomposed approach applied to a large-scale system of vehicles, for example, a 250-vehicle system (Section 4.1) and a 1,000-vehicle system (Section 4.2), respectively.

4.1 Numerical experiments on a 250-vehicle system

To test the performance of our decomposed approach, we considered a 250-vehicle system with five speeds on the Chicago highway and grid networks. In this case, the number of binary variables and constraints for the entire CPMS model is still manageable. For the undecomposed approach, we ran Gurobi on the entire 250-vehicle CPMS instance for one hour. For the decomposed approach, we used Algorithm 1 to divide 250 vehicles into six similarity groups with the maximum group size 50, and ran each for 10 minutes (60 minutes/6 groups) for each group independently and summed their siz objective values. Results are shown in Figures 4(a)–4(b).

Note that for the two networks, the decomposed approach gives slightly better objective values than does the undecomposed approach when $P = 0.1 \sim 0.4$. In contrast, the undecomposed approach slightly outperforms the decomposed approach for the grid-network instances at $P = 1.2 \sim 2.0$. This result occurs because the arrival time constraints are not binding for large P, thereby reducing the complexity of solving the entire problem. In such cases, the undecomposed approach identifies intercluster platooning opportunities, helping further reduce fuel consumption; these are opportunities unavailable to the decomposed approach.

4.2 Numerical experiments on a 1,000-vehicle system

We also compared performance of the two approaches on a 1,000-vehicle system with the five speed settings on the both networks. For the undecomposed approach, we still limited Gurobi to one hour of computational time. For the decomposed approach, we partitioned the 1,000 vehicles into 25 similarity groups with a maximum group size of 60. The CPMS instances for each group were given 2.4 minutes (60 minutes/25 groups) of computational time. The performance of both methods is shown in Figures 4(c)–4(d). Note that the undecomposed approach is much less effective at solving such large problem instances: the relative gap between the two approaches ranges from 12.5% to 30%. The undecomposed approach also fails to estimate a lower bound on fuel consumption. Furthermore, the fuel consumption given by the undecomposed approach is not monotonically decreasing as P increases. These observations indicate that a system of 1,000 vehicles is beyond the limits of problems that can be addressed by the undecomposed approach. Furthermore, the performance of the two methods indicates that a decomposed approach can obtain a reasonable suboptimal solution for problem instances with a large number of vehicles.

5 Conclusion and Discussion

The CPMS model improves the model from Larson et al. (2016) in that vehicles can traverse edges at different speeds; this more accurately models the real world. This freedom also creates opportunities for fuel savings in addition to those offered by platooning. We also propose a clustering algorithm to extend the applicability of our platooning model by decomposing large-scale problems into independent subproblems. Our numerical experiments show that in little computational time, the decomposed approach can find much better solutions than the undecomposed approach can when applied to a large set of vehicles distributed in a complex transport network. Although the undecomposed approach can outperform the decomposed approach, their relative differences are small in problem instances with a few hundred vehicles.

We are currently improving our vehicle platooning model in order to address important real-world situations. Work includes extending the model to accounting for traffic flow and congestion by allowing the cost of traversing an edge to change over time. We are also interested in understanding the practicality of forming platoons under various traffic flow conditions. These improvements require incorporating governing equations of traffic flow in the model, which inevitably complicate the model; care will be necessary in order to ensure that problem instances remain tractable.

To further improve the decomposed approach for large-scale problems, we believe a two-phase optimization strategy will be beneficial. In the first phase, subproblems will solved independently as was done in Section 4. The second phase will involve fixing vehicle routes obtained from the first phase but enabling schedule coordination in order to allow for more interclustering platooning opportunities.

Acknowledgements

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357.

References

References

Alam, A.A., Gattami, A., Johansson, K.H., 2010. An experimental study on the fuel reduction potential of heavy duty vehicle platooning, in: 13th International IEEE Conference on Intelligent Transportation Systems, IEEE. doi:`10.1109/itsc.2010.5625054`.

Auld, J., Hope, M., Ley, H., Sokolov, V., Xu, B., Zhang, K., 2016. POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations. Transportation Research Part C: Emerging Technologies 64, 101–116. doi:10.1016/j.trc.2015.07.017.

Baskar, L., Hellendoorn, J., Schutter, B.D., Papp, Z., 2011. Traffic control and intelligent vehicle highway systems: A survey. IET Intelligent Transport Systems 5, 38–52. doi:10.1049/iet-its.2009.0001.

Baskar, L.D., Schutter, B.D., Hellendoorn, H., 2013. Optimal routing for automated highway systems. Transportation Research Part C: Emerging Technologies 30, 1–22. doi:10.1016/j.trc.2013.01.006.

Bonnet, C., Fritz, H., 2000. Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing, in: SAE Technical Paper Series, SAE International. doi:10.4271/2000-01-3056.

Browned, F., McArthur, J., Radovich, C., 2004. Fuel saving achieved in the field test of two tandem trucks. California PATH Research Report UCB-ITS-PRR-2004-20. University of Southern California.

Chan, C.C., 2007. The state of the art of electric, hybrid, and fuel cell vehicles. Proceedings of the IEEE 95, 704–718. doi:10.1109/jproc.2007.892489.

Chang, C.S., Liao, W., Chen, Y.S., Liou, L.H., 2016. A mathematical theory for clustering in metric spaces. IEEE Transactions on Network Science and Engineering 3, 2–16. doi:10.1109/tnse.2016.2516339.

Daimler Innovation, . Connected trucks: Freight transport of the future by using the internet. URL: https://www.daimler.com/innovation/connectivity/connected-trucks.html.

Fernandes, P., Nunes, U., 2012. Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow. IEEE Transactions on Intelligent Transportation Systems 13, 91–106. doi:10.1109/tits.2011.2179936.

GAMS Development Corporation, 2016. General Algebraic Modeling System (GAMS) Release 24.7.4. URL: http://www.gams.com.

Ghasemi, A., Kazemi, R., Azadi, S., 2015. Exact stability of a platoon of vehicles by considering time delay and lag. Journal of Mechanical Science and Technology 29, 799–805. doi:10.1007/s12206-015-0142-x.

Gurobi Optimization, Inc., 2017. Gurobi optimizer reference manual. URL: http://www.gurobi.com.

Harrington, W., Krupnick, A., 2012. Improving fuel economy in heavy-duty vehicles. Resources for the Future DP 12-02 .

Hobert, L.H.X., 2012. A study on platoon formations and reliable communication in vehicle platoons. Master's thesis. University of Twente.

Kammer, C., 2013. Coordinated heavy truck platoon routing using global and locally distributed approaches. M.S. Thesis. KTH Royal Institute of Technology.

Larson, J., Liang, K.Y., Johansson, K.H., 2015. A distributed framework for coordinated heavy-duty vehicle platooning. IEEE Transactions on Intelligent Transportation Systems 16, 419–429. doi:10.1109/tits.2014.2320133.

Larson, J., Munson, T., Sokolov, V., 2016. Coordinated platoon routing in a metropolitan network, in: 2016 Proceedings of the Seventh SIAM Workshop on Combinatorial Scientific Computing, Society for Industrial and Applied Mathematics. pp. 73–82. doi:10.1137/1.9781611974690.ch8.

Larsson, E., Sennton, G., Larson, J., 2015. The vehicle platooning problem: Computational complexity and heuristics. Transportation Research Part C: Emerging Technologies 60, 258–277. doi:10.1016/j.trc.2015.08.019.

Li, S.E., Li, K., Wang, J., 2013. Economy-oriented vehicle adaptive cruise control with coordinating multiple objectives function. Vehicle System Dynamics 51, 1–17. doi:10.1080/00423114.2012.708421.

Liang, K.Y., 2014. Coordination and routing for fuel-efficient heavy-duty vehicle platoon formation. Licentiate Thesis. KTH.

Liang, K.Y., 2016. Fuel-efficient heavy-duty vehicle platoon formation. Ph.D. thesis. KTH School of Electrical Engineering.

Liang, K.Y., Mårtensson, J., Johansson, K.H., 2013. When is it fuel efficient for a heavy duty vehicle to catch up with a platoon? Elsevier BV. pp. 738–743. doi:10.3182/20130904-4-jp-2042.00071.

Luo, X.Y., Shladover, S.E., 2011. Automated Truck Platoon Control. Technical Report UCB-ITS-PRR-2011-13. University of California, Berkeley. California PATH Research Report.

Navigant Research, 2014. Automotive fuel efficiency technologies. URL: https://www.navigantresearch.com/research/automotive-fuel-efficiency-technologies.

Nowakowski, C., Shladover, S.E., Cody, D., 2011. Cooperative adaptive cruise control: Testing drivers' choice of following distances. California PATH Research Report UCB-ITS-PRR-2011-01. University of California, Berkeley.

Pollet, B.G., Staffell, I., Shang, J.L., 2012. Current status of hybrid, battery and fuel cell electric vehicles: From electrochemistry to market prospects. Electrochimica Acta 84, 235–249. doi:10.1016/j.electacta.2012.03.172.

Shida, M., Nemoto, Y., 2009. Development of a small-distance vehicle platooning system, in: 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services.

The International Energy Agency, 2014. Key world energy statistics. URL: http://www.fossilfuelsreview.ed.ac.uk/resources/Evidence%20-%20Climate%20Science/IEA%20-%20Key%20World%20Energy%20Statistics.pdf.

U.S. Department of Energy, 2011. How you can save over 30% on gas this summer. URL: https://energy.gov/articles/how-you-can-save-over-30-gas-summer. July 6, 2011.

U.S. Department of Energy, n.d. Driving more efficiently. URL: http://www.fueleconomy.gov/feg/driveHabits.jsp. retrieved June 19, 2017.

U.S. Energy Information Administration, 2015. Use of energy in the United States explained. URL: http://www.eia.gov/energyexplained.

U.S. Environment Protection Agency, 2014. Sources of greenhouse gas emissions. URL: https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions.

van de Hoef, S., Johansson, K.H., Dimarogonas, D.V., 2015. Coordinating truck platooning by clustering pairwise fuel-optimal plans, in: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, IEEE. doi:10.1109/itsc.2015.75.

Wang, L.Y., Syed, A., Yin, G., Pandya, A., Zhang, H., 2012. Coordinated vehicle platoon control: Weighted and constrained consensus and communication network topologies, in: 51st Conference on Decision and Control, IEEE. doi:10.1109/cdc.2012.6427034.